Business Tools

- Document Handling
  - Install Stirling PDF

Document Handling

Document Handling

Install Stirling PDF

https://www.youtube.com/embed/o8N-njA57iQ

PDF is a realisty of business, just as Fax machines remain a primary method of communication for banks, hospitals, and yes, even criminal justice information systems (int eh United States anyway). If you've never had to sign a form from someone then return it...only to realize you needed to print the form, then sign it, then re-scan it into your machine and send it back, then you are a lucky person indeed.

Not to worry, since PDF is still a business staple these days, we have incredible open source tools like Libre Office, Open Office, and Only Office that allow us to draft and create documents of all kinds, and then export them as PDF (a much less useful format for sure). Why do we do this? We hate ourselves I suppose. Ha!

Seriously though, PDF is a reality, and dealing with them can be tedious at best unless you've been willing to invest in the Adobe paid products that allow you to edit and do all kind so interesting things with the document. If you weren't then you could read them at least.

Now, we have Stirling PDF which will let you do all kinds of powerful things to PDF documents. You can convert them into rich text, word documents (libre office open document text), power point presentations, and much more. You can merge, split, and even use tools to add a "wet" signature to a PDF.

Let's get Stirling PDF installed and running. You can install it locally if you prefer, but four our purposes, we'd rather set it up on a server (it runs as a web application), and allow our staff to put it to use as well.

What you'll need

- A machine you wish to host Stirling PDF on
- Docker-CE and Docker Compose installed on the machine
- About 15 minutes of your time

Installation

Installation via a Simple Script

You can easily install Docker-CE, Docker-Compose, Portainer-CE, and NGinX Proxy manager by using this quick install script I created and maintain on Github. Just use the command:

`wget -O install-docker.sh  https://gitlab.com/bmcgonag/docker_installs/-/raw/main/install_docker_nproxyman.sh`

To download the script to your desired host.

Change the permissions to make the script executable:

`chmod +x ./install-docker.sh`

and then run the script with the command:

`./install-docker.sh`

When run, the script will prompt you to select your host operating system, then will ask you which bits of software you want to install.

Simply enter 'y' for each thing you want to install.

At some point, you may be asked for your super user (sudo) password as well.

Allow the script to complete installation.

At this point, you might want to log out and back in, as this will allow you to use the `docker` and `docker-compose` commands without the need of 'sudo' in front of them.

Install Stirling PDF

Let's quickly create a new folder for our Stirling PDF install.

`mkdir -p docker/stirling-pdf`

We'll move into the folder, and create a new file called "docker-compose.yml":

`cd docker/stirling-pdf`

`nano docker-compose.yml`

Now we need to copy and paste the contents below into that file:

```
---
services:
  stirling-pdf:
    image: frooodle/s-pdf:latest
    ports:
```

```
      - '8080:8080'   # change the left side port if 8080 is already in use on your host
machine
    volumes:
      - ./trainingData:/usr/share/tessdata #Required for extra OCR languages
      - ./extraConfigs:/configs
#      - /location/of/customFiles:/customFiles/
#      - ./logs:/logs/
    environment:
      - DOCKER_ENABLE_SECURITY=false
      - INSTALL_BOOK_AND_ADVANCED_HTML_OPS=false
      - LANGS=en_US    # change this to your preferred language code
```

In the above code, we need to do a few things. First, we'll create the volumes (folders) for our volume mappings. This will keep all of the data na configuration for Stirling PDF in this single folder, which makes it much easier to backup.

```
mkdir -p {trainingData,extraConfigs,logs}
```

Next, we may need to change the left side of the port mapping. This mapping tells docker that the port 8080 on our host server should be mapped to port 8080 in our container. If the port on our host server is already in use by another application, we can change the port number on the left side of the colon ':' to a port that is not in use on our host. In the video, I changed this to port 8092. If 8080 is not in use on your host, then no need to change this number.

Next, you may want to change the language code on the last line to your preferred language. I'm in the US, so I have it set to 'en_US'.

Once you've finished making any needed changes, save the file with CTRL + O, then press Enter to confirm, and exit the nano editor with CTRL + X.

Start the Application

Now we are ready to start our application. We can do this by combining two command into a one-liner:

```
docker compose up -d && docker compose logs -f
```

This single line command will bring down the docker images we need for Stirling PDF, and will start the containers running, then will show us the running log output of the containers as they start. All we are really looking for in the logs are any errors. We can address those as they come up. If you've set everything properly, you shouldn't have any issues.

Once the logs slow down or stop rolling, you can use CTRL + C to setop seeing the logs, and get back to a normal command prompt.

Now, we are ready to check out our Stirling PDF application.

Open your favorite modern web browser, and go to the IP address of your host, and the port number you entered on the left side of the port mapping.

```
http://192.168.10.23:8080
```

Of course, replace the IP and port with the correct IP of your host server and port number you entered in the docker compose file.

You should be presented with the Stirling PDF home screen. From here you can search for functions, or scroll through the list of capabilities. The power is now yours to manipulate PDFs like a pro!

Check out my video for a few simple use cases.

Support My Channel and Content

Support my Channel and ongoing efforts through Patreon:

https://www.patreon.com/awesomeopensource

Buy me a Beer / Coffee:
https://paypal.me/BrianMcGonagill